

## Übungen zum Kapitel 2

### Grundlagen in Python und Jupyter Notebooks

Erstellt und überarbeitet: armin.baenziger@zhaw.ch, 2. Januar 2020

```
In [1]: %autosave 0
```

```
Autosave disabled
```

**(A.1)** Versuchen Sie, diesen Satz **genau so** mit Markdown *in der folgenden Zelle* zu replizieren.

Versuchen Sie, diesen Satz **\*\*genau so\*\*** mit Markdown *\*in der folgenden Zelle\** zu replizieren.

**(B.1)** Erzeugen Sie die Liste `Liste`, welche die ersten vier grossen Buchstaben im Alphabet enthält.

```
In [2]: Liste = ['A', 'B', 'C', 'D'] # oder: Liste = list('ABCD')
```

**(B.2)** Weisen Sie den dritten Buchstaben der Liste `Liste` der Variable `s` zu.

```
In [3]: s = Liste[2]
```

**(B.3)** Ersetzen Sie in `Liste` das zweite Element (`B`) durch den Buchstaben `Z`.

```
In [4]: Liste[1] = 'Z'
```

**(B.4)** Sortieren Sie `Liste` mit der `sort`-Methode.

```
In [5]: Liste.sort()  
Liste
```

```
Out[5]: ['A', 'C', 'D', 'Z']
```

**(C.1)** Berechnen Sie in Python  $1.0325^5$ .

```
In [6]: 1.0325 ** 5
```

```
Out[6]: 1.1734113958293944
```

**(C.2)** Das letzte Ergebnis wird jeweils im Objekt `_` zwischengespeichert. Runden Sie damit das letzte Resultat auf zwei Nachkommastellen wie folgt:

```
round(_, 2)
```

```
In [7]: round(_, 2)
```

```
Out[7]: 1.17
```

Für die nächsten Aufgaben generieren wir zuerst die drei Listen `A`, `B` und `C`:

```
In [8]: A = list(range(5))
        B = [0, 1, 2, 3, 4]
        C = [1, 0, 7, 3, 5]
        print('A =', A)
        print('B =', B)
        print('C =', C)
```

```
A = [0, 1, 2, 3, 4]
B = [0, 1, 2, 3, 4]
C = [1, 0, 7, 3, 5]
```

**(D.1)** Prüfen Sie mit `is` ob `A` und `B` auf das gleiche Objekt verweisen.

```
In [9]: A is B
```

```
Out[9]: False
```

**(D.2)** Prüfen Sie mit `==`, ob die zwei Listen `A` und `B` gleich sind.

```
In [10]: A == B
```

```
Out[10]: True
```

**(D.3)** Prüfen Sie mit `==`, ob die zwei Listen `B` und `C` gleich sind.

```
In [11]: B == C
```

```
Out[11]: False
```

**(D.4)** Prüfen Sie, ob das letzte Element von `B` dem letzten Element von `C` entspricht.

```
In [12]: B[-1] == C[-1]
```

```
Out[12]: False
```

**(D.5)** Prüfen Sie mit `==`, ob die ersten zwei Elemente von `B` und `C` identisch sind.

```
In [13]: B[:2] == C[:2]
```

```
Out[13]: False
```

**(D.6)** Prüfen Sie, ob je das zweite *und* je das vierte Element der Listen `B` und `C` identisch ist.

```
In [14]: (B[1] == C[1]) & (B[3] == C[3])
```

```
Out[14]: False
```

**(D.7)** Prüfen Sie, ob je das zweite *oder* je das vierte Element der Listen `B` und `C` identisch ist.

```
In [15]: (B[1] == C[1]) | (B[3] == C[3])
```

```
Out[15]: True
```

**(D.8)** Prüfen Sie mit `len`, ob die zwei Listen `B` und `C` gleich viele "Elemente" (Items) enthalten.

```
In [16]: len(B) == len(C)
```

```
Out[16]: True
```

Für die nächsten Fragen erzeugen wir zunächst zwei Strings:

```
In [17]: s1 = 'Datenanalyse'      # geht mit '
        s2 = "mit Python"        # oder mit "
```

**(E.1)** Zählen Sie mit der `count`-Methode die Anzahl *n* im String `s1`.

```
In [18]: s1.count('n')
```

```
Out[18]: 2
```

**(E.2)** Erzeugen Sie aus den Strings `s1`, `s2` und einem Leerschlag den String `'Datenanalyse mit Python'`. Weisen Sie das Ergebnis dem Objekt `s` zu. Tipp: Verwenden Sie zum Verknüpfen `+`.

```
In [19]: s = s1 + ' ' + s2
        s
```

```
Out[19]: 'Datenanalyse mit Python'
```

**(E.3)** "Slicen" Sie die ersten 10 Zeichen aus `s`.

```
In [20]: s[:10]
```

```
Out[20]: 'Datenanaly'
```

---

**(F.1)** Erzeugen Sie mit `range()` und `list()` eine Liste, welche wie folgt aussieht: `[5, 7, 9, 11, 13, 15, 17, 19]`

```
In [21]: list(range(5, 20, 2))
```

```
Out[21]: [5, 7, 9, 11, 13, 15, 17, 19]
```

**(F.2)** Berechnen Sie die Quadratwurzeln der Werte in der unten definierten Liste `listli`. Verwenden Sie hierzu eine `for`-Schleife.

```
In [22]: listli = [9, 25, 3, 16, 100]
for wert in listli:
    print(wert ** 0.5)
```

```
3.0
5.0
1.7320508075688772
4.0
10.0
```

**(F.3)** Nutzen Sie eine `for`-Schleife mit `if`-Verzweigung, um alle Elemente der Liste `listli` auszugeben, welche grösser als 10 sind.

```
In [23]: for wert in listli:
        if wert > 10:
            print(wert)
```

```
25
16
100
```

---

**Ende der Übung**